IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 30 May 2024, accepted 12 June 2024, date of publication 1 July 2024, date of current version 6 August 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3421571

RESEARCH ARTICLE

Hybrid Bat and Salp Swarm Algorithm for Feature Selection and Classification of Crisis-Related Tweets in Social Networks

NAFEES AKHTER FAROOQUI[1], MOHAMMAD KAMRUL HASAN[2], (Senior Member, IEEE), MOHAMMED AHSAN RAZA NOORI[3], ABDUL HADI ABD RAHMAN[2], SHAYLA ISLAM[4], (Senior Member, IEEE), MOHAMMAD HALEEM[5], SHEIKH FAHAD AHMAD[6], ASIF KHAN[7], (Member, IEEE), FATIMA RAYAN AWAD AHMED[8], NISSREIN BABIKER MOHAMMED BABIKER[9], THOWIBA E. AHMED[10], AND ATTA UR REHMAN KHAN[11], (Senior Member, IEEE)

[1]Department of CSE, Koneru Lakshmaiah Education Foundation, Vaddeswaram, Andhra Pradesh, India
[2]Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia, Bangi 43600, Malaysia
[3]School of Computing, DIT University, Dehradun, India
[4]Institute of Computer Science and Digital Innovation, UCSI University, Kuala Lumpur 56000, Malaysia
[5]Department of Computer Science, Era University, Lucknow, India
[6]Department of Computer Science and Engineering, GITAM Deemed to be University, Hyderabad, India
[7]Department of Computer Science, Integral University, Lucknow, India
[8]Computer Science Department, College of Computer Engineering and Science, Prince Sattam Bin Abdulaziz University, Al-Kharj 16273, Saudi Arabia
[9]Computer Science Department, College of Science and Humanities-Jubail, Imam Abdulrahman Bin Faisal University, Dammam 35811, Saudi Arabia
[10]Bisha—Information System Department, College of Computing and Information Technologies, University of Bisha, Bisha 61922, Saudi Arabia
[11]College of Engineering and IT, Ajman University, Ajman, United Arab Emirates

Corresponding authors: Mohammad Kamrul Hasan (hasankamrul@ieee.org) and Asif Khan (asif05amu@gmail.com)

This work was supported by the Universiti Kebangsaan Malaysia Under GUP 2023-010.

ABSTRACT Twitter is a useful tool for effectively tracking and managing crisis-related incidents. However, due to many irrelevant features in textual data, the problem of high dimensionality arises, which eventually increases the computational cost and decreases classification performance. Thus, to handle such a problem, this work presents a Spark-based hybrid binary Bat (BBA) and binary Salp swarm algorithm (BSSA) named SBBASSA for feature selection and classification of crisis-related tweets. In the proposed technique, the hybridization of standard BBA and BSSA algorithms is performed to enhance their exploration capabilities, then the combined algorithm is implemented in parallel using Apache Spark framework to reduce the overall execution time during the feature selection process. A support vector machine (SVM) classifier is applied during the wrapper-based feature subset selection and classification. The performance of the proposed SBBASSA was analyzed on six benchmark crisis tweet datasets, namely Hurricane Sandy, Boston Bombings, Oklahoma Tornado, West Texas Explosion, Alberta Floods, and Queensland Floods, and then compared with standard BSSA, BBA, and binary particle swarm optimization (BPSO). Results showed that SBBASSA performed competently in the feature selection and classification, outperformed other algorithms in crisis tweet classification, and achieved the highest accuracy with the lowest feature set in a reduced execution time.

INDEX TERMS Apache spark, bat algorithm, crisis tweet classification, feature selection, salp swarm algorithm.

I. INTRODUCTION

Social media platforms play a vital role in quickly understanding casualties during disasters. Several studies have discovered that the general public uses social media applications

The associate editor coordinating the review of this manuscript and approving it for publication was Maria Chiara Caschera.

during disasters to provide information about urgent needs, damage to infrastructure, people injured or dead, volunteer or charity efforts, and situational updates. Hence, timely access to social media data can be leveraged for emergency response in the first few hours to reduce both human loss and economic damage significantly [4], [7], [14], [15], [16]. However, one of the main challenges in utilizing social media in crisis management is the reliable detection of useful messages from massive textual data. Tweet classification for disaster response is a text classification task that determines the class of tweets from predefined informative and not-informative classes [14]. Still, due to irrelevant and redundant features in the available disaster dataset, the performance of machine learning classifiers suffers a lot.

To handle such problems, feature selection techniques can be employed to select the most important features and remove the irrelevant ones. Based on the feature selection algorithms, the most informative and relevant features are selected, improving the classification performance of disaster tweets. As a feature selection technique, nature-inspired meta-heuristic algorithms emerged as a popular method for selecting optimal features and improving classification performance due to their high robustness and efficiency in exploiting and exploring the vast feature space [17].

The Bat algorithm (BA) is a novel evolutionary optimization algorithm proposed by Yang in [2], inspired by the echolocation behavior of bats. Bats use echolocation to navigate and hunt for prey, emitting ultrasound pulses and adjusting their flight paths based on the echoes they receive. Similarly, virtual bats search for optimal solutions in BA by adjusting their positions in the search space based on echolocation-inspired rules. Due to its simple concepts, fast convergent speed, and ease of implementation, BA has been applied successfully in many fields [37]. On the other hand, the Salp swarm algorithm (SSA) [18] is also a recently developed meta-heuristic optimization algorithm that imitates the food-searching behavior of sea salps, which are gelatinous marine animals. In SSA, salps collectively search for optimal solutions by mimicking the swimming behavior of real salp swarms. Its simplicity is combined with robustness and flexibility. The SSA proved to be effective in dealing with different optimization and feature selection problems, and it has been successfully utilized in a wide range of problems in different fields, such as machine learning, engineering design, wireless networking, image processing, and power energy [19]. Hence, motivated by the adeptness of BA and SSA to perform optimal feature selection and enhance the classification of crisis-related tweets, this work presents SBBASSA, a hybridization of BA and SSA to enhance the searchability/exploration of feature space, and Apache Spark-based parallel implementation to reduce the execution time. The primary motivation behind this approach is the need for effective analysis of social media data during disasters, which can provide valuable insights for crisis management. Traditional methods often struggle with social media data's high dimensionality and noise. To address these challenges, SBBASSA leverages the hybridization of BBA and BSSA to enhance feature selection capabilities. The parallel implementation using Apache Spark aims to reduce execution time during feature selection. SBBASSA optimizes feature subsets for classification performance by integrating a wrapper-based approach with SVM. The rationale is to provide a robust and efficient tool for analyzing large-scale social media data, and improving decision-making in disaster management scenarios. The following are the major contributions of this work:

1) SBBASSA, hybridization of BA with SSA and its Apache Spark-based parallel implementation is proposed for feature selection.
2) SBBASSA was employed to perform feature selection and classification of six crisis tweet datasets, including Hurricane Sandy, Boston Bombings, Oklahoma Tornado, West Texas Explosion, Alberta Floods, and Queensland Floods.
3) The performance of SBBASSA was compared with several algorithms including BBA, BSSA, and BPSO.

The rest of this article is ordered as follows: The literature review is discussed in Section II. Section III presents the background concepts related to BBA, BSSA and Apache Spark. Section IV discusses the methodology and proposed SBBASSA. Section V presents the experimental results and discussion. Section VI presents the comparative analysis. Lastly, Section VII concludes the work and future direction.

II. LITERATURE REVIEW

The literature contains significant studies on Twitter about crisis detection. Researchers proposed different approaches for crisis tweet classification using machine learning, deep learning and feature selection through meta-heuristic algorithms. In [4], [40] to categorize tweets amid disaster, the authors employed a convolutional neural network (CNN) and proposed a graph-based semi-supervised learning technique. During the initial phase of the disaster, they experimented with unlabeled data from the Queensland floods and Nepal earthquake, then compared the results without adding unlabeled data. Study [3], [41] categorized social media posts on disasters based on their informativeness using machine learning so that informative posts can lead to better disaster management services. The proposed method utilized a random forest (RF) classifier to identify fire, earthquake, and flood-related content. Reference [5] designed a cross-domain classifier based on linguistic, emotional, and sentimental elements from the messages related to disasters. They conducted in-depth experiments using 26 different disaster-related datasets. They used 25 datasets for training and one for testing in each experiment. On the other hand, [6] provides a framework for monitoring, analyzing, and classifying social media posts according to their relevance. Additionally, the study provides explicit and abstract rules for classifying relevant social media posts during disasters. The authors evaluate data using RF for classification and combine social media metadata into a batch learning technique. Furthermore, they provide a method for early relevance classification evaluation that incorporates active, incremental, and online learning to lower the number of labeled data needed and to fix algorithmic misclassifications through feedback classification.

The authors in [7] applied term frequency-inverse document frequency (TF-IDF) and machine learning classifiers, including K-nearest neighbor (KNN), RF, NB, support vector machine (SVM), and gradient boosting (GB) to detect tweets and estimate the significant harm during disasters. Based on crisis word embeddings and glove embeddings, they also implemented deep learning approaches like CNN, long-short-term memory (LSTM), gated recurrent unit (GRU), Bi-directional GRU, and GRU-CNN. Various disaster datasets are analyzed, and the best-performing results of machine learning and deep learning techniques are compared. Similarly, [8] worked on examining the content of tweets and classifying them by the different types of emergencies. For instance, in a crisis event affecting crucial infrastructures, they verified if a tweet had words associated with infrastructures such as roads. When a pipeline explosion disaster occurred in Ludwigshafen, Germany, they gathered 3785 tweets about it. They used supervised learning algorithms like NB, Decision tree (DT), RF, SVM, and neural network (NN) to determine whether a tweet is related to the accident or not. The goal of [9] is to mitigate the generalization limitation of tweet-wise pre-trained models during automatic identification and semantically compilation of tweets relevant to crises. Such models were used with unsupervised semantic tweet clustering to address this limitation. Both classification and clustering of tweets were accomplished using latent space characterizations of tweets based on sentence encoders. Researchers in [21] released Disaster Tweet Corpus 2020, one of the most comprehensive collections of tweets on disasters, which comprised 48 disasters of 9 different types, and performed an extensive evaluation of CNN, bidirectional encoder representations from transformers (BERT), and universal sentence encoder (USE) in cross-disaster and cross-type settings. Likewise [14] investigated the performance of best-performing machine learning and deep learning models for classifying disaster tweets in three settings: in-disaster, out-disaster, and cross-disaster. The Bi-LSTM model with WordVec gives the best results in all three experimental settings.

In addition to improving the classification performance, recent studies incorporated meta-heuristic algorithms for feature selection. Reference [10] introduced a framework to monitor tweets amid natural disasters and classify them based on the need for or availability of general or medical resources and any location information (if any) provided. Various statistical classifiers were used to demonstrate their effectiveness for greater results, and a forest optimization-based wrapper feature selection algorithm was employed to enhance classification performance. The framework is tested using FIRE, SMERP, and CrisisLex datasets, and its performance for efficient resource management is shown. The results of the experiments show that a multinomial NB classifier wrapped in a forest optimization algorithm (FOA) performs well and has a very short execution time. To classify crisis events, [11] presents a hybrid system combining the pre-trained DistilBERT feature extraction model with binary hunger games search for feature selection. The proposed approach outperforms feature selection and event identification methods regarding feature reduction and accuracy. The classification and mapping of tweets linked to disasters were performed in [12] using a genetic algorithm (GA)-based feature selection approach. Features are extracted from tweets to form a matrix of feature vectors. Experiments revealed the merit of feature selection using GA to lower feature space while enhancing the classification of disaster-related tweets. In [13] a framework for improving the classification of tweets related to natural disasters has been put forth. An improved GA was employed for feature selection, disaster risk reduction, and response in the Philippines. The algorithm aims to identify the most relevant features from a vast feature space to classify disaster tweets. Moreover [17] presents an Apache Spark-based parallel BGWO and BWOA algorithm for optimal feature selection and classification of disaster tweets. RF is applied for wrapper-based feature selection and classification. The performance of proposed optimizers was evaluated on seven datasets of CrisisMMD. Results showed that both optimizers performed competently in the feature selection and classification.

Several works in the literature employed machine learning and deep learning techniques to classify disaster tweets; on the contrary, few works have investigated the application of population-based meta-heuristic algorithms in disaster tweet analysis to handle the problem of high dimensionality. Hence, to solve the problem, this work utilizes the binary variant of a recently developed BA and SSA technique and proposes two enhancements named SBBASSA. The BBA hybridized with BSSA to enhance its feature space searchability and then its parallel implementation is done using the Apache Spark framework to reduce the execution time. The experiments were performed on six crisis datasets, including Hurricane Sandy, the Boston Bombings, the Oklahoma Tornado, the West Texas Explosion, the Alberta Floods, and the Queensland Floods.

III. RESEARCH METHODS AND MATERIALS

A. BAT ALGORITHM

The BA has been inspired by the echolocation behavior of bats [2]. Bats use natural sonar to do this. The two main characteristics of bats when finding prey have been adopted in designing the BA. Bats tend to decrease the loudness and increase the rate of emitted ultrasonic sound when they chase prey. This behavior has been mathematically modeled as follows: In the BA, an artificial bat has a position vector, velocity vector, and frequency vector, which are updated during iterations as (1), (2), and (3):

$$V_i(t+1) = V_i(t) + (X_i(t) - best)F_i \tag{1}$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{2}$$

where *best* is the best solution attained so far and F_i indicates the frequency of i-th bat which is updated in each course of iteration as follows

$$F_i = F_{min} + (F_{max} - F_{min})\beta \tag{3}$$

where β is a random number of a uniform distribution in[0,1]. It is clear from the Eqs. (1) and (3) that different frequencies encourage artificial bats to have a diverse propensity to the best solution.

These equations could guarantee the exploitability of the BA. However, a random walk procedure is also has been used to perform the exploitation as follows:

$$X_{new} = X_{old} + \varepsilon A^t \tag{4}$$

In this formula, ε is a random number in [−1,1], and A is the loudness of emitted sound that bats use to perform an exploration instead of exploitation as it is increased. Thus, it can be stated that BA is a balanced combination of PSO and intensive local search. The loudness (A) and pulse emission rate (r) control the balancing between these techniques. These two elements are updated as follows:

$$A_i(t+1) = \alpha A_i(t) \tag{5}$$

$$r_i(t+1) = r_i(0)[1 - exp(-\gamma t)] \tag{6}$$

where α and γ are constants. Eventually, A_i will equal zero, while the final value of r_i is r(0). Note that both loudness and rate are updated when the new solutions are improved to guarantee that the bats are moving toward the best solutions.

The pseudocode of BA is presented in algorithm 1.

Algorithm 1 Pseudo-Code of BA

Initialize the bat population
Define pulse frequency F_i
Initialize pulse rates r_i and the loudness A_i
while t<*maxIteration* **do**
    Adjusting frequency and updating velocities positions using equations (1 to 3)
    if ($rand > r_i$) **then**
        Select a solution (best) among the best solutions randomly
        Change some of the dimensions of the position vector with some of the dimensions of best
    end if
    Generate a new solution by flying randomly
    if ($rand < Ai\&f(xi) < f(best)$) **then**
        Accept the new solutions
        Increase r_i and reduce A_i
    end if
    Rank the bats and find the current best
end while

B. SALP SWARM ALGORITHM

SSA [18] is another meta-heuristic algorithm motivated by the swarming nature of salps in the deep ocean foraging for food. Salps have small barrel-shaped, translucent, and gelatinous bodies that navigate in the ocean through self-propulsion. The individual salps attach and form a spiral shape called a salp chain, as shown in Figure. 1. The salps move together in search of food sources. A typical salp chain generally contains a leader and multiple followers salps where the leader guides the followers and changes his position in the direction of food source [18]. Mathematically, the leader updates its position according to equation (7)

$$x_j^1 = \begin{cases} F_j + c_1((ub_j - lb_j)c_2 + lb_j) & c_3 \geq 0 \\ F_j - c_1((ub_j - lb_j)c_2 + lb_j) & c_3 < 0 \end{cases} \tag{7}$$

where, x_j^1 represents leader, F_j is the position of food source, ub_j and lb_j represents upper and lower search space boundaries, respectively and c_1, c_2, and c_3 are random numbers $\in$ [0, 1].

However, $c1$ is regarded as the most crucial parameter as it regulates the balance between exploration and exploitation,

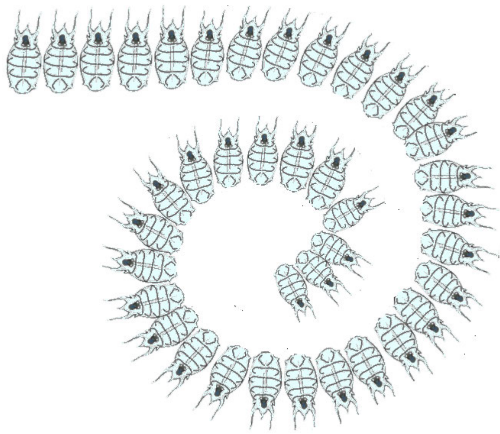


FIGURE 1. The salp chain.

which influences the efficiency of the search process overall. The equation for $c1$ is:

$$c_1 = 2e^{-(4l/L)^2} \tag{8}$$

where, l and L are the current iteration and total number of iterations, respectively.

The position of follower salps are updated according to Newton's law of motion and represented as:

$$x_j^i = \frac{1}{2}at^2 + v_0 t \tag{9}$$

where, x_j^i is the ith follower, $i \geq 2$. $a = \frac{v_{final}}{v_0}$, $v = \frac{x - x_0}{t}$, v_0 is initial speed, and t denotes time.

In the optimization process, time is substituted by iteration, the gap between iterations is equal to 1, and considering

$v_0 = 0$, equation (9) becomes:

$$x_j^i = \frac{1}{2}(x_j^i + x_j^{i-1}) \tag{10}$$

The pseudo-code of SSA is shown in Algorithm 2.

C. BINARY BAT AND SALP SWARM ALGORITHM

The standard BA and SSA are designed to solve problems with continuous values. However, feature selection problems are binary, where continuous values are transformed into possible discrete values of 0 and 1, where 0 means the feature is not selected and 1 is selected. In BBA and BSSA, a transformation function transforms the values from continuous to discrete. The Sigmoid function is a popular transformation function that is widely used to perform these kinds of transformations. Its mathematical formula is shown in equation (11)

$$S(x_j^i) = \frac{1}{1 + e^{-x_j^i}} \tag{11}$$

where x_j^i is jth element of ith population

Finally, whether a particular feature will be selected or omitted from the feature set is determined by equation (12)

$$X_i^j = \begin{Bmatrix} 0 \ if \quad rand < S(x_i^j) \\ 1 \quad\quad otherwise \end{Bmatrix} \tag{12}$$

Algorithm 2 Pseudo-Code SSA

```
Initialize the salp population
while t<maxIteration do
    Evaluate fitness of salps
    F = best salp
    Update c1 by Eq. (8)
    for each salp do
        if (i == 1) then
            Update leader salp by Eq. (7)
        else
            Update follower salps by Eq. (10)
        end if
    end for
end while
return F
```

D. APACHE SPARK

The Apache Spark platform is an open-source cluster computing platform that can be used for distributed computing of all types. A driver program performs parallel computations on a cluster in a primary-secondary architecture. Spark's fundamental data abstraction is Resilient Distributed Datasets (RDDs). It is a read-only data collection partitioned into multiple worker nodes in a cluster [17]. Spark offers in-memory computation, which enables data and preliminary outcomes to be retained in memory, avoiding input-output delay caused by transferring data back and forth from the hard drive [31].

IV. PROPOSED SBBASSA FOR FEATURE SELECTION

In the proposed algorithm, the hybridization of standard BBA and BSSA algorithms is performed to enhance their exploration capabilities further; then the combined algorithm is implemented in parallel using the Apache Spark framework to reduce the overall execution time during the feature selection process.

After the initialization of the population, the fitness of each agent is evaluated, which gives the initial global best solution. The algorithm then enters into an actual optimization/feature selection phase where BBA is executed first in every iteration, and then global fitness is also evaluated and updated if it is better than the current best solution. The algorithm then moves toward the BSSA module and updates the position of leader and followers, after that fitness is calculated, and the global best solution is updated. The same feature selection process is repeated over a specified number of iterations whose final output is the set of optimal features.

SBBASSA, the parallel implementation of the integrated BBA and BSSA in the Apache Spark environment is presented as a pseudo-code in Algorithm 3. In line 1 of SBBASSA, the agents population with their binary positions is initialized by calling Spark's *parallelize* method creating an RDD *pos*. In line 2, the fitness of each agent is computed. The *map* method is applied on *pos* to compute the fitness and *collect* method is called to return the fitness with corresponding positions in *pos_fitness* in the driver node. In line 3, broadcast variable *posBC* is created using Spark's *broadcast* to cache the current position of agents to every worker node. In line 4, from the *pos_fitness* variable, the best solution having minimum fitness value is obtained and saved inside the *best* variable. Furthermore, in line 5, the best solution *best* is broadcasted and referenced in *bestBC* for future use.

This concludes the initialization of population, fitness evaluation, and ranking of the global best agent based on fitness. In terms of feature selection, the first subset of features is generated. To get the optimal feature subset, SBBASSA moves to the optimization phase by executing a loop where various operations are executed until the maximum number of iterations is reached.

In line 7 of SBBASSA, using the position value of broadcasted variable *posBC* and *bestBC* the *pos_bat* is updated and distributed using *parallelize* method. In line 8, the fitness of each agent in *pos_bat* is calculated, and the *collect* method is called to return the fitness with corresponding positions in bat_fitness in the driver node. In line 9, from the bat_fitness best solution having minimum fitness value is obtained and saved inside *tmpfitness* variable. Then, from the line (12-16), the fitness values of agents in bat_fitness and *pos_fitness* are compared to replace the agents in *pos_fitness* if the agents in bat_fitness have a better fitness value. In line 17, the updated *pos_fitness* is broadcasted and referenced in *posBC* for future use. From lines (18-22), the *tmpFitness* will decide whether it will replace the current global best solution based on the new fitness value.

Algorithm 3 Pseudo-Code of SBBASSA

```
Input: maxIter:total no. of iterations, m:numPartitions, n:population size, d:dimension
Output: optimal feature set
1: pos ← sc.parallelize((n × d), m)                         ▷ Initialize binary population at random [0 or 1]
2: pos_fitness ← pos.map(getFitness).collect()                ▷ Using Eq. (13)
3: posBC ← sc.broadcast(list(map(lambda x : x[1], pos_fitness)))
4: best ← min(pos_fitness, key=itemgetter(0))
5: bestBC ← sc.broadcast(best)
6: while t > maxIter do
7:     pos_bat ← sc.parallelize(range(n), m).map(updatebat)   ▷ Using Eq. (1 - 3), and (11)
8:     bat_fitness ← pos_bat.map(getfitness).collect()        ▷ Using Eq. (13)
9:     tmpFitness ← min(bat_fitness, key=itemgetter(0))
10:    vel ← pos_bat.map(lambda x: x[1]).collect()
11:    velBC ← sc.broadcast(vel)
12:    for i in range(n) do
13:        if (bat_fitness[i][0] ≤ pos_fitness[i][0]and(random.random() < 0.5) then
14:            pos_fitness[i]=bat_fitness[i]
15:        end if
16:    end for
17:    posBC ← sc.broadcast(list(map(lambda x : x[1], pos_fitness)))
18:    if tmpFitness[0] < bestBC.value[0] then
19:        best ← tmpFitness
20:        bestBC.destroy();
21:        bestBC ← sc.broadcast(best)
22:    end if
23:    pos_ssa ← sc.parallelize(range(n), m).map(updateSSA)   ▷ Using Eq. (7), (8), (10), and (11)
24:    ssa_fitness ← pos_ssa.map(getfitness).collect()        ▷ Using Eq. (13)
25:    posBC ← sc.broadcast(list(map(lambda x : x[1], ssa_fitness)))
26:    tmpFitness ← min(ssa_fitness, key=itemgetter(0))
27:    if tmpFitness[0] < bestBC.value[0] then
28:        best ← tmpFitness
29:        bestBC.destroy()
30:        bestBC = sc.broadcast(best)
31:    end if
32:    t = t + 1
33: end while
34: return best
```

In line 23 of SBBASSA, using the position value of broadcasted variable *posBC* and *bestBC*, the *pos_ssa* is updated and distributed using the *parallelize* method. In line 24, the fitness of each agent in *pos_ssa* is calculated, and the *collect* method is called to return the fitness with corresponding positions in *ssa_fitness* in the driver node. In line 25, the *ssa_fitness* is broadcasted and referenced in *posBC* for future use. In line 26, from the ssa_fitness, the best solution having minimum fitness value is obtained and saved inside the *tmpfitness* variable. From lines (27-31), the *tmpFitness* will decide whether it will replace the current global best solution based on the new fitness value. Then, the loop continues to iterate until stop criteria are not met, and finally, the SBBASSA returns the optimal feature subset. A wrapper-based feature selection using SVM is implemented in this work. The feature selection process is performed during the fitness evaluation of the agent population. The fitness function is shown in equation (13).

$$Fitness = \alpha(1 - a) + \beta\frac{|R|}{|C|} \tag{13}$$

where a is the accuracy, R is the size of the feature subset, and C is the total number of features. $\alpha \in [0, 1]$ and $\beta = 1 - \alpha$.

V. METHODOLOGY

This section presents the overall workflow designed to carry out the feature selection and classification of disaster tweets, including a detailed explanation of the proposed SBBASSA algorithm. The major steps are data collection, preprocessing, feature extraction, feature selection, and classification as shown in Figure 2.

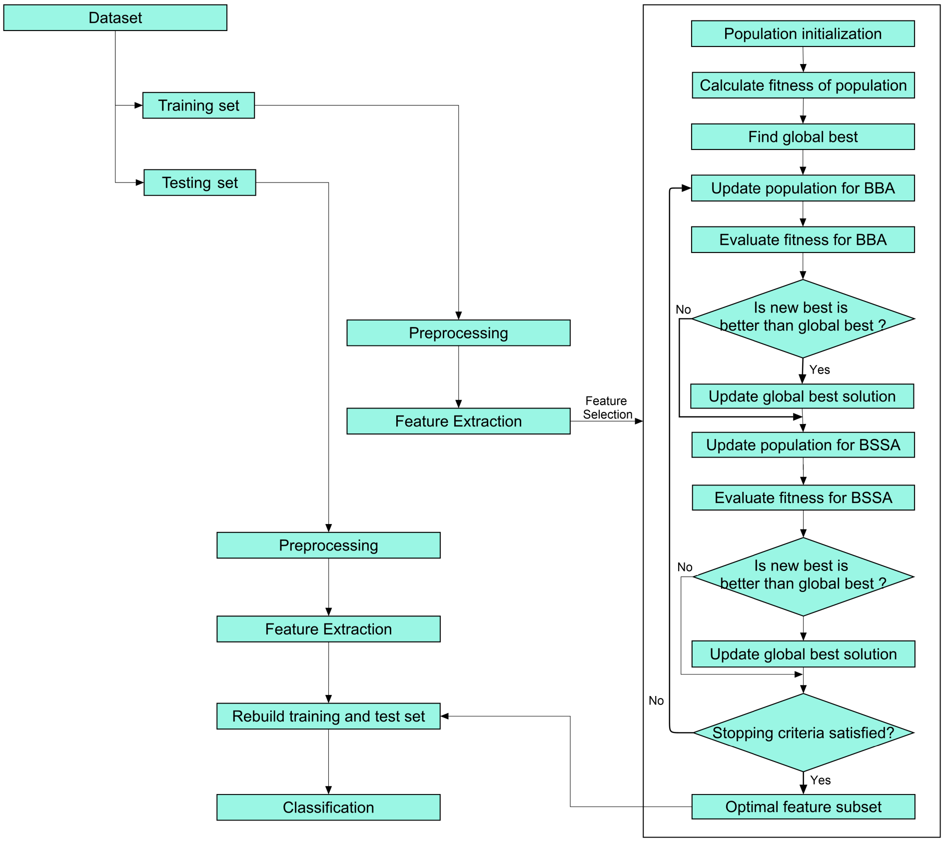


FIGURE 2. **Workflow of the proposed methodology.**

TABLE 1. **Summary of datasets.**

Datasets	On-topic	Off-topic	Total
Sandy Hurricane	6138	3871	10009
Alberta Floods	5189	4843	10032
Boston Bombings	5648	4365	10013
Oklahoma Tornado	4827	5166	9993
Queensland Floods	5414	4620	10034
West Texas Explosion	5246	4761	10007

A. DATASETS

We used six different datasets from CrisisLex.org to show their applicability and compare them against existing approaches on these crisis-related datasets. The corpus CrisisLexT6 [32] includes English tweets for 6 crisis events in 2012 and 2013: Hurricane Sandy, the Boston Bombings, the Oklahoma Tornado, the West Texas Explosion, the Alberta Floods, and the Queensland Floods. About 60,000 tweets ($\sim$10,000 per crisis event) are labeled by crowdsourcing workers, whether the tweet is related "on-topic" or not related "off-topic" to a particular crisis event. Table 1 shows the details related to the datasets.

B. PREPROCESSING

Tweets in general contain a bunch of raw and noisy data that is not very beneficial in the text mining process, so preprocessing of disaster tweets is required to clean and morph the data for relevant feature extraction [29]. This study employed the following steps for the preprocessing of disaster tweets:

- *Lower casing* of text in the tweets.
- *Remove* URL, words inside angular and square brackets, @username, special characters, standardize the words, non-ASCII characters, and additional white spaces.
- *Stopwords* filtering such as pronouns and prepositions.
- *Remove* words less than two length.
- *Stemming* to retain the root word.
- *Remove* tweets containing less than three words.
- *Remove* less frequent words in the corpus.

The preprocessing steps applied in this study are general for any dataset retrieved from the Twitter social media platform.

C. FEATURE EXTRACTION

The internal structure of textual data is so unordered it cannot be straightforwardly fed into machine learning classifiers. A feature extraction approach is needed to change the data into a structured and numerical form [30]. In this work, TF-IDF is utilized to extract features from disaster tweets. It is based on a statistical weighting mechanism frequently used for term weighting in text mining-related fields.

TABLE 2. Parameter setting for all algorithms.

Algorithms	Parameters with their values
BBA	$A = 0.5, r = 0.5, Qmin = 0, Qmax = 2$
BSSA	$c1$ *and* $c2$ *random numbers over* $[0, 1]$
BPSO	$c1 = 2, c2 = 2, Vmax = 6, wMin = 0.9, wMax = 0.2$
SBBASSA	$A = 0.5, r = 0.5, Qmin = 0, Qmax = 2, c1$ *and* $c2$ *random numbers over* $[0, 1]$
Parameters for all	*No. of search agents* $= 7$, *No. of iterations* $= 100$, *Population dimension* $=$ *No. of features in data*, *Search domain* $= [0, 1]$

TABLE 3. Performance comparison of RF, NB, and SVM.

Datasets	RF				NB				SVM			
	A	P	R	F	A	P	R	F	A	P	R	F
Sandy Hurricane	90.31	90.83	90.43	90.62	78.87	76	96.2	84.9	**90.91**	90.86	90.63	**90.74**
Alberta Floods	93.97	91.57	97.3	94.35	88.14	84.36	94.61	89.19	**94.72**	94.21	95.66	**94.93**
Boston Bombings	90.01	90.99	90.62	90.8	85.72	85.23	90.35	87.71	**90.66**	90.42	92.49	**91.44**
Oklahoma Tornado	90.1	89.82	90.82	90.31	89.99	89.16	90.27	89.71	**90.4**	88.82	93.03	**90.87**
Queensland Floods	96.06	96.92	95.75	96.33	91.48	92.3	91.87	92.09	**96.46**	97.92	95.48	**96.68**
West Texas Explosion	94.95	95.96	94.19	95.06	94.16	92.6	96.57	94.55	**95.3**	96.89	96	**96.44**

TABLE 4. Performance comparison of BSSA, BBA, BPSO, and SBBASSA.

Datasets	BSSA				BBA				BPSO				SBBASSA			
	A	P	R	F	A	P	R	F	A	P	R	F	A	P	R	F
Sandy Hurricane	90.3	93.04	90.76	91.88	90.37	92.03	90.33	91.17	89.52	89	90.99	89.98	**93.77**	92.51	92.77	**92.64**
Alberta Floods	94.57	94.24	96.15	95.18	95.61	95.88	96.62	96.24	94.37	93.63	96.67	95.12	**96.61**	96.94	96.15	**96.54**
Boston Bombings	90.93	90.61	90.05	90.32	91.29	93.45	90.9	92.15	91.61	90.64	94.02	92.29	**94.75**	95.01	94.12	**94.56**
Oklahoma Tornado	91.1	92.75	90.62	91.67	91.95	92.46	93.33	92.89	90.74	91.55	92.1	91.82	**95.25**	94.1	95.28	**94.68**
Queensland Floods	95.32	96.73	94.9	95.8	96.47	97.44	95.9	96.66	95.18	96.3	95	95.64	**97.51**	97.2	98.82	**98**
West Texas Explosion	96.65	96.2	96.48	96.33	96.36	97.27	98.38	97.82	96.33	95.78	97.08	96.42	**98.53**	98.22	98.03	**98.12**

D. FEATURE SELECTION

Feature selection is the process of choosing an optimal subset of features from a dataset with many features. It helps to reduce data dimensionality and improve the performance of classification algorithms. When using a metaheuristic algorithm as a wrapper-based feature selection technique, the agent population is typically represented in binary form (1, 0). Each feature corresponds to a binary digit in the population, where 1 indicates selection and 0 indicates exclusion [17]. In this study, we proposed SBBASSA for feature selection from disaster tweets. Its entire detail is discussed in Section IV.

E. CLASSIFICATION

In this subtask, the optimal feature subsets generated from the feature selection process are used to train and test the classifier for final classification. In this study, we employed SVM to train and test crisis-related datasets.

VI. EXPERIMENTAL RESULTS AND DISCUSSION

This section presents the findings of different experiments performed on six crisis datasets: Hurricane Sandy, Boston Bombings, Oklahoma Tornado, West Texas Explosion, Alberta Floods, and Queensland Floods, gathered from [35]. The BSSA, BBA, BPSO, and, SBBASSA are employed on the datasets for feature selection. The best results are represented in bold values. The datasets were divided into 80%

Sequential algorithms such as BSSA, BBA, and BPSO are implemented in Python language. In contrast, SBBASSA is implemented in PySpark using RDD API of Apache Spark framework and datasets are broadcasted to each worker node. NLTK, Numpy, and Sklearn libraries perform various natural language processing, matrix manipulation, and classification tasks. All experiments are conducted on the Google cloud platform for the SBBASSA YARN cluster with 4 instances of virtual machines of type n1-standard-2 with 2 vCPU and 7.5 GB memory each is created, and for sequential algorithms, 1 instance of virtual machine of type n1-standard-2 with 2 vCPU and 15 GB memory is created. The parameter settings for all algorithms are shown in Table 2. A seed value of 42 is set to address the randomness of the stochastic algorithms.

A. RESULTS

The choice of SVM for feature selection and classification is not random. We performed a comparative analysis between RF, SVM, and NB, where SVM emerged as the best classifier, as seen in Table 3 and Figure 3. The accuracy and f-score of RF are close to SVM but inferior. The performance of NB is way lower than SVM and RF.

Table 4 and Figure 4, illustrate the SBBASSA's classification performance in contrast to the regular BSSA and other optimization methods such as BBA, and BPSO. It is clear from the results that SBBASSA has achieved better accuracy and f-score on all 6 datasets than other optimizers.

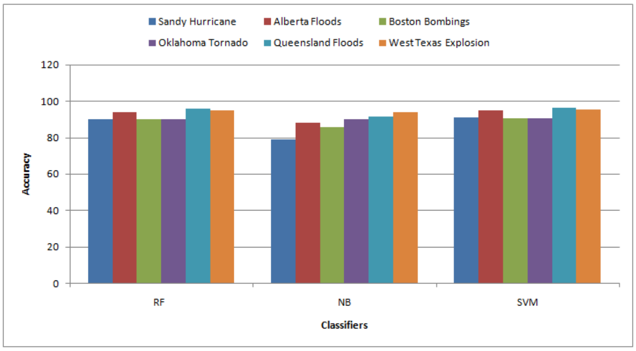


FIGURE 3. Accuracy comparison of RF, NB, and SVM.

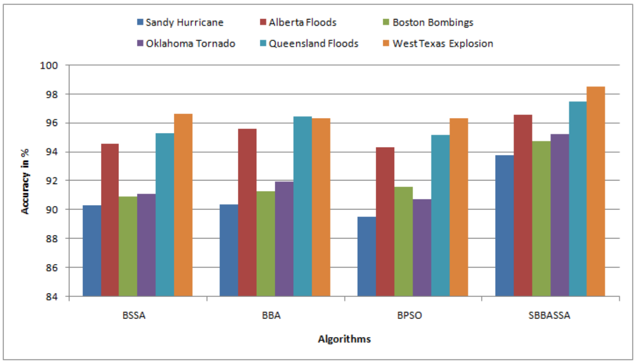


FIGURE 4. Accuracy comparison of BSSA, BBA, BPSO, and SBBASSA.

TABLE 5. Features selected by BSSA, BBA, BPSO, and SBBASSA.

Datasets	Initial Features	BSSA	BBA	BPSO	SBBASSA
Sandy Hurricane	2742	1735	1677	1708	**1629**
Alberta Floods	3704	2122	1938	2063	**1722**
Boston Bombings	2767	1745	1675	1793	**1339**
Oklahoma Tornado	2691	1753	1690	1713	**1625**
Queensland Floods	2832	1809	1677	1755	**1552**
West Texas Explosion	2455	1630	1291	1638	**1199**

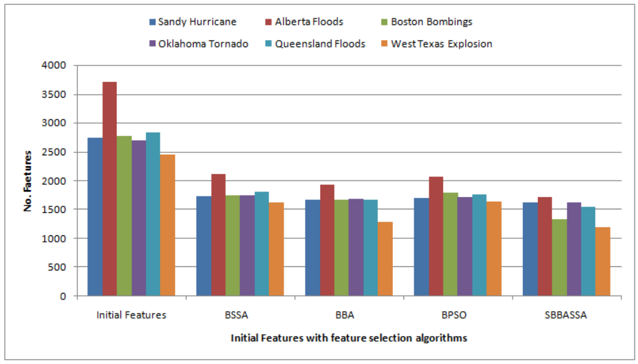


FIGURE 5. Comparison of initial Features vs features selected by BSSA, BBA, BPSO, and SBBASSA.

TABLE 6. Execution time (in seconds) of SBBASSA with other algorithms.

Datasets	BSSA	BBA	BPSO	SBBASSA
Sandy Hurricane	5527	4623	4563	**3027**
Alberta Floods	5805	4923	4716	**3115**
Boston Bombings	5689	4741	4806	**3011**
Oklahoma Tornado	5027	4569	4713	**3132**
Queensland Floods	5474	4607	4665	**3192**
West Texas Explosion	5416	4880	4861	**3185**

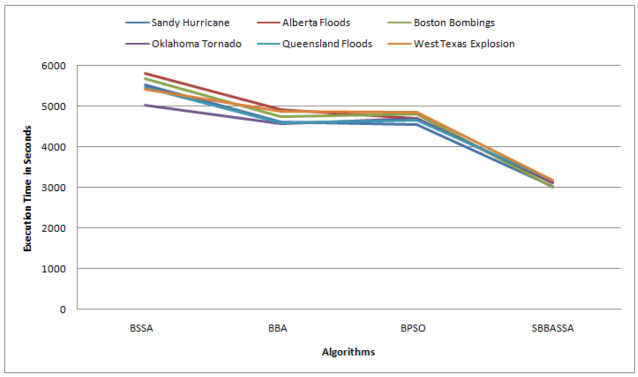


FIGURE 6. Comparison between the execution time of BSSA, BBA, BPSO, and SBBASSA.

By hybridizing the BBA and BSSA, the SBBASSA can prevent local optima and enhance the diversity of solutions. These results demonstrate the robustness of SBBASSA's capacity to balance the exploitation and exploration while selecting the optimal features that yield better classification results.

Table 5 and Figure 5 show feature selection results of different optimizers, and the SBBASSA selected minimum features compared to other feature selection algorithms used in this study. During the process, in SBBASSA every agent in BBA interacts with a random agent and updates its position to explore the new potential solutions that help in optimal feature selection. Hence, it is worth noticing that the hybridization of BBA with BSSA has enhanced its searchability to explore the relevant features from high-dimensional datasets deeply.

The comparison of the execution time of competing algorithms is shown in Table 6 and Figure 6. It can be observed that for each of the six datasets, SBBASSA costs much less time than all other algorithms. SBBASSA is designed to execute in a parallel and distributed environment, reducing its overall execution time. Hence, it can be inferred that the proposed SBBASSA is a suitable algorithm for solving the feature selection and classification task from high-dimensional crisis datasets in less time. In general, the computational complexity of a population-based meta-heuristic optimizer is dependent on the size of the population (p), dimension (d), and maximum number of iterations (i), so the time complexity of the sequential algorithm is $O(p * d * i)$. On the other hand, the time complexity of SBBASSA can be estimated by considering the number of nodes in the cluster (n), the number of cores in each node (c), and communication overhead on parallelizing the job (j). Hence, the resulting time complexity of SBBASSA is $O(((p * d * i)/n * c) + j)$.

TABLE 7. Comparison of SBBASSA with previous studies.

Research	Year	Method	Accuracy
[1]	2019	Word2Vec-LR	92.30
[1]	2019	Word2Vec-SVM	91.24
[1]	2019	Word2Vec-NN	93.65
[33]	2019	Glove-CNN	91.3
[36]	2022	Word2Vec-LSTM	91.7
Proposed	Present	SBABSSA	**96.07**

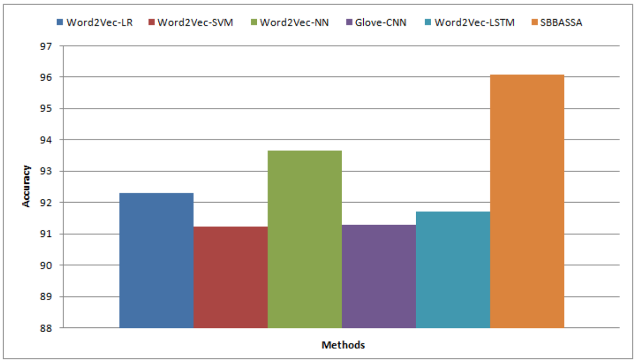


FIGURE 7. Performance comparison between SBBASSA and previous works.

To verify the efficiency of the proposed SBBASSA, we compared it with the prior studies. The comparison is shown in Table 7 and Figure 7, where SBBASSA outperformed the results of previous works. From the comparison, it can be inferred that the proposed SBBASSA is suitable for feature selection and classification of disaster-related tweets.

B. DISCUSSION

BSSA is predominantly an exploitation-focused algorithm through its leader-followers dynamics. The leader navigates towards the best solution, and the following salps move based on the position of their predecessor, allowing for diverse exploration of the search space. BSSA adjusts the movement step size dynamically, enabling salps to follow the leader closely and venture into new areas, thus avoiding local optima and encouraging broader search capabilities. Additionally, there is an element of randomness in the leader's movement, which introduces variability and helps the salps to explore new potential areas within the feature space. By integrating these exploration features of BSSA with the echolocation-inspired exploration tactics of the BBA, our hybrid SBBASSA approach achieves a more balanced search strategy, enhancing its ability to explore and exploit the feature space effectively.

The primary focus on hybridizing BBA and BSSA algorithms is to leverage their complementary strengths, enhancing feature space exploration and exploitation. Both have complementary strengths that synergize well in feature selection tasks. BBA's echolocation behavior encourages the exploration of diverse feature subsets, while SSA's social interactions facilitate convergence towards promising regions. By combining these strengths, SBBASSA harnesses the benefits of both algorithms, resulting in enhanced feature selection capabilities for crisis data analysis.

In the wrapper-based feature selection process during each iteration, SBBASSA generates a subset of features evaluated by SVM through classification. This interaction leads to a synergistic feedback loop between feature selection and classification, resulting in optimal features and improved classification performance. It is worth mentioning that the feature selection process in SBBASSA is not inherently tied to any specific classifier. Instead, it combines BBA and BSSA to search for an optimal feature subset that maximizes classification performance. This approach allows SBBASSA to explore the feature space comprehensively, identifying features relevant for classification across various models, not just SVM.

On the other hand, the scalability of the SBBASSA algorithm in a distributed computing environment is facilitated by its parallel implementation using Apache Spark, which is ensured through effective load balancing and fault tolerance mechanisms. Regarding load balancing, Apache Spark dynamically distributes tasks across worker nodes in the cluster, ensuring that the workload, which primarily involves evaluating feature subsets and updating solution spaces, is evenly distributed. This is achieved through automatic data partitioning and dynamic resource allocation, preventing any single node from being overwhelmed with excessive processing tasks. Additionally, Spark's ability to adjust resource allocation based on the workload helps maintain optimal performance throughout the feature selection process. Regarding fault tolerance, Apache Spark employs Resilient Distributed Datasets (RDDs) and lineage graphs to reconstruct lost data partitions during node failures. SBBASSA can also utilize task redundancy, checkpointing, and speculative execution to enhance fault tolerance. These mechanisms ensure that SBBASSA can efficiently process large volumes of data and maintain robustness, making it well-suited for scalable feature selection mechanisms.

VII. CONCLUSION

In this study, our contribution lies in the development of SBBASSA, an Apache Spark-based parallel implementation of hybrid BBA and BSSA is proposed to find the right subset of features from large feature space and improve the classification process of crisis-related tweets in less time. The novelty of SBBASSA lies in its parallel and hybrid approach, which leverages the complementary strengths of BBA and BSSA, leading to improved convergence properties and enhanced feature selection performance in a reduced execution time. Wrapper-based SBBASSA with SVM is leveraged for feature selection and classification. The SBBASSA was applied to six crisis datasets, including Hurricane Sandy, the Boston Bombings, the Oklahoma Tornado, the West Texas Explosion, the Alberta Floods, and the Queensland Floods. Results were compared with the binary version of SSA, BA, and PSO. Experimental results showed that the proposed SBBASSA outperformed the other algorithms regarding classification, number of selected features, and execution time. Furthermore, on comparing the results with the previous studies, the SBBASSA outperformed them in terms of accuracy. Overall, SBBASSA represents a significant advancement in feature selection, offering a valuable tool for real-time crisis management systems and other applications requiring effective analysis of textual data.

Future research can examine the potential of SBBASSA to solve problems, including sentiment analysis, intrusion detection, and other high-dimensional feature space problems. The integration of BA with other meta-heuristic algorithms can also be investigated. In addition, SBBASSA can be integrated into existing crisis management systems to enhance their capabilities in processing and analyzing social media data during disasters. Challenges include addressing data quality and noise, ensuring scalability, maintaining interoperability, addressing ethical and privacy concerns, and ensuring model interpretability. Solutions involve robust filtering techniques, optimization for scalability, standardization of data formats, implementation of privacy-preserving techniques, and development of model interpretation methods. By addressing these considerations, SBBASSA can enhance the responsiveness and effectiveness of real-time crisis management systems in mitigating the impacts of natural disasters.

ACKNOWLEDGMENT
This work was supported by the Universiti Kebangsaan Malaysia Under the research grant scheme GUP 2023-010.

CONFLICTS OF INTEREST
The authors declare there is no conflict of interest.

REFERENCES
[1] S. Manna and H. Nakai, "Effectiveness of word embeddings on classifiers: A case study with tweets," in *Proc. IEEE 13th Int. Conf. Semantic Comput. (ICSC)*, Jan. 2019, pp. 158–161.

[2] X.-S. Yang, "A new metaheuristic bat-inspired algorithm," in *Nature Inspired Cooperative Strategies For Optimization (NICSO)*. Berlin, Germany: Springer, 2010, p. 6574.

[3] F. S. Acerbo and C. Rossi, "Filtering informative tweets during emergencies: A machine learning approach," in *Proc. 1st CoNEXT Workshop ICT Tools Emergency Netw. Disaster Relief*, 2017, p. 16.

[4] F. Alam, S. Joty, and M. Imran, "Graph based semi-supervised learning with convolution neural networks to classify crisis related tweets," in *Proc. 12th Int. AAAI Conf. Web Social Media*, 2018, pp. 1–4.

[5] D. Graf, W. Retschitzegger, W. Schwinger, B. Pröll, and E. Kapsammer, "Cross-domain informativeness classification for disaster situations," in *Proc. 10th Int. Conf. Manage. Digit. EcoSystems*, Sep. 2018, Art. no. 183190.

[6] M.-A. Kaufhold, M. Bayer, and C. Reuter, "Rapid relevance classification of social media posts in disasters and emergencies: A system and evaluation featuring active, incremental and online learning," *Inf. Process. Manage.*, vol. 57, no. 1, Jan. 2020, Art. no. 102132.

[7] A. Kumar, J. P. Singh, and S. Saumya, "A comparative analysis of machine learning techniques for disaster-related tweet classification," in *Proc. IEEE R10 Humanitarian Technol. Conf. (R10-HTC)*, Nov. 2019, pp. 222–227.

[8] M. Habdank, N. Rodehutskors, and R. Koch, "Relevancy assessment of tweets using supervised learning techniques: Mining emergency related tweets for automated relevancy classification," in *Proc. 4th Int. Conf. Inf. Commun. Technol. Disaster Manage. (ICT-DM)*, Dec. 2017, pp. 1–8.

[9] J. Kersten, J. Bongard, and F. Klan, "Combining supervised and unsupervised learning to detect and semantically aggregate crisis-related Twitter content," in *Proc. Anal. Detection Models Disaster-Related Tweets*, 2021, pp. 744–754.

[10] A. Bhoi, R. C. Balabantaray, D. Sahoo, G. Dhiman, M. Khare, F. Narducci, and A. Kaur, "Mining social media text for disaster resource management using a feature selection based on forest optimization," *Comput. Ind. Eng.*, vol. 169, Jul. 2022, Art. no. 108280.

[11] H. Adel, A. Dahou, A. Mabrouk, M. Abd Elaziz, M. Kayed, I. M. El-Henawy, S. Alshathri, and A. Amin Ali, "Improving crisis events detection using DistilBERT with hunger games search algorithm," *Mathematics*, vol. 10, no. 3, p. 447, Jan. 2022.

[12] I. P. Benitez, A. M. Sison, and R. P. Medina, "Implementation of GA-based feature selection in the classification and mapping of disaster-related tweets," in *Proc. 2nd Int. Conf. Natural Lang. Process. Inf. Retr.*, Sep. 2018, p. 16.

[13] I. P. Benitez, A. M. Sison, and R. P. Medina, "An improved genetic algorithm for feature selection in the classification of disaster-related Twitter messages," in *Proc. IEEE Symp. Comput. Appl. Ind. Electron. (ISCAIE)*, Apr. 2018, pp. 238–243.

[14] N. Algiriyage, R. Prasanna, K. Stock, R. Sampath, E. E. H. Doyle, and D. Johnston, "Identifying disaster-related tweets: A large-scale detection model comparison," in *Proc. Int. ISCRAM Conf.*, May 2021, pp. 731–743.

[15] R. Alrashdi and S. O'Keefe, "Deep learning and word embeddings for tweet classification for crisis response," 2019, *arXiv:1903.11024*.

[16] S. Madichetty and M. Sridevi, "Detecting informative tweets during disaster using deep neural networks," in *Proc. 11th Int. Conf. Commun. Syst. Netw. (COMSNETS)*, Jan. 2019, pp. 709–713.

[17] M. A. R. Noori, B. Sharma, and R. Mehra, "Feature selection from disaster tweets using spark-based parallel meta-heuristic optimizers," *Social Netw. Anal. Mining*, vol. 12, no. 1, p. 118, Dec. 2022.

[18] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, vol. 114, pp. 163–191, Dec. 2017.

[19] L. Abualigah, M. Shehab, M. Alshinwan, and H. Alabool, "Salp swarm algorithm: A comprehensive survey," *Neural Comput. Appl.*, vol. 32, no. 15, pp. 11195–11215, Aug. 2020.

[20] M.-Y. Cheng and D. Prayogo, "Symbiotic organisms search: A new meta-heuristic optimization algorithm," *Comput. Struct.*, vol. 139, pp. 98–112, Jul. 2014.

[21] M. Wiegmann, J. Kersten, F. Klan, M. Potthast, and B. Stein, "Analysis of detection models for disaster-related tweets," in *Proc. ISCRAM Conf. 17th Int. Conf. Inf. Syst. Crisis Response Manage.*, 2020, pp. 872–880.

[22] M. Imran, C. Castillo, J. Lucas, P. Meier, and S. Vieweg, "AIDR: Artificial intelligence for disaster response," in *Proc. 23rd Int. Conf. World Wide Web*, Apr. 2014, Art. no. 159162.

[23] A. Olteanu, C. Castillo, F. Diaz, and S. Vieweg, "CrisisLex: A lexicon for collecting and filtering microblogged communications in crises," in *Proc. 8th Int. AAAI Conf. Weblogs Social Media*, 2014, pp. 376–385.

[24] A. Olteanu, S. Vieweg, and C. Castillo, "What to expect when the unexpected happens: Social media communications across crises," in *Proc. 18th ACM Conf. Comput. Supported Cooperat. Work Social Comput.*, Feb. 2015, Art. no. 9941009.

[25] M. Imran, P. Mitra, and J. Srivastava, "Enabling rapid classification of social media communications during crises," *Int. J. Inf. Syst. Crisis Response Manage.*, vol. 8, no. 3, pp. 1–17, Jul. 2016.

[26] F. Alam, F. Ofli, and M. Imran, "CrisisMMD: Multimodal Twitter datasets from natural disasters," in *Proc. 12th Int. AAAI Conf. Web Social Media*, 2018, pp. 465–473.

[27] K. Stowe, M. Palmer, J. Anderson, M. Kogan, L. Palen, K. M. Anderson, R. Morss, J. Demuth, and H. Lazrus, "Developing and evaluating annotation procedures for Twitter data during hazard events," in *Proc. Joint Workshop Linguistic Annotation, Multiword Expressions Constructions (LAW-MWE-CxG-2018)*, 2018, pp. 133–143.

[28] A. J. Mcminn, Y. Moshfeghi, and J. M. Jose, "Building a large-scale corpus for evaluating event detection on Twitter," in *Proc. 22nd ACM Int. Conf. Conf. Inf. Knowl. Manage. (CIKM)*, 2013, Art. no. 409418.

[29] M. A. R. Noori and R. Mehra, "Fire emergency detection from Twitter using supervised principal," in *Proc. IEEE 15th Int. Conf. Ind. Inf. Syst. (ICIIS)*, Nov. 2020, pp. 403–408.

[30] M. A. R. Noori and R. Mehra, "Traffic congestion detection from Twitter using Word2Vec," in *ICT Analysis and Applications*. Singapore: Springer, 2021, pp. 527–534.

[31] H. Ahmed, E. M. Younis, A. Hendawi, and A. A. Ali, "Heart disease identification from patients' social posts, machine learning solution on spark," *Future Gener. Comput. Syst.*, vol. 111, pp. 714–722, Oct. 2020.

[32] A. Olteanu, "CRISISLEX: A lexicon for collecting and filtering microblogged communications in crises," in *Proc. Int. AAAI Conf. Web Social Media*, 2014, vol. 8, no. 1, pp. 376–385.

[33] J. Kersten, A. Kruspe, M. Wiegmann, and F. Klan, "Robust filtering of crisis-related tweets," in *Proc. ISCRAM Conf. 16th Int. Conf. Inf. Syst. Crisis Response Manage.*, 2019, pp. 814–824.

[34] J. Liu, T. Singhal, L. T. M. Blessing, K. L. Wood, and K. H. Lim, "CrisisBERT: A robust transformer for crisis classification and contextual crisis embedding," in *Proc. 32st ACM Conf. Hypertext Social Media*, Aug. 2021, pp. 133–141.

[35] *CrisisLexT6*. Accessed: Feb. 6, 2022. [Online]. Available: https://www.crisislex.org/data-collections.html#CrisisLexT6

[36] A. Kumar, J. P. Singh, Y. K. Dwivedi, and N. P. Rana, "A deep multi-modal neural network for informative Twitter content classification during emergencies," *Ann. Operations Res.*, vol. 319, no. 1, pp. 791–822, Dec. 2022.

[37] S. Q. A. Rizvi, G. Wang, A. Khan, M. K. Hasan, T. M. Ghazal, and A. U. R. Khan, "Classifying Parkinson's disease using resting state electroencephalogram signals and u EN-PDNet," *IEEE Access*, vol. 11, pp. 107703–107724, 2023.

[38] A. Khan, J. P. Li, and M. A. Husain, "Power grid stability analysis using pipeline machine," *Multimedia Tools Appl.*, vol. 82, no. 17, pp. 25651–25675, Jul. 2023.

[39] A. Khan, J. P. Li, N. Ahmad, S. Sethi, A. U. Haq, S. H. Patel, and S. Rahim, "Predicting emerging trends on social media by modeling it as temporal bipartite networks," *IEEE Access*, vol. 8, pp. 39635–39646, 2020.

[40] A. Khan, M. K. Hasana, T. M. Ghazal, S. Islam, H. M. Alzoubi, U. A. Mokhtar, R. Alam, and M. Ahmad, "Collaborative learning assessment via information and communication technology," in *Proc. RIVF Int. Conf. Comput. Commun. Technol. (RIVF)*, Dec. 2022, pp. 311–316.

[41] A. Khan, J. P. Li, M. K. Hasan, N. Varish, Z. Mansor, S. Islam, R. A. Saeed, M. Alshammari, and H. Alhumyani, "PackerRobo: Model-based robot vision self supervised learning in CART," *Alexandria Eng. J.*, vol. 61, no. 12, pp. 12549–12566, Dec. 2022.

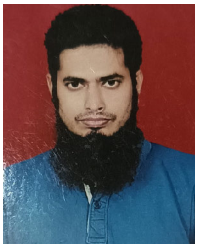
NAFEES AKHTER FAROOQUI received the B.Sc. degree (Hons.) in statistics from AMU, Aligarh, the M.C.A. degree from Integral University, Lucknow, and the Ph.D. degree in computer science and engineering from DIT University, Dehradun, Uttarakhand, India. He has 13 years of teaching experience. He is currently an Assistant Professor with the Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur, Andhra Pradesh, India. His research interests include artificial intelligence, machine learning, deep learning, computer vision, pattern recognition, natural language processing, and data mining. He has published approx. 30 papers in international journals and conference proceedings, including Web of Science and Scopus indexed journals. He has also published various book chapters in Scopus-indexed journals. He has published two books for national publishers. He has been a member of the International Association of Engineers (IAENG), since 2017, and ACM, since 2011. He guided various projects of UG and PG level students. He received awards from various professional bodies.

MOHAMMAD KAMRUL HASAN (Senior Member, IEEE) received the Doctor of Philosophy (Ph.D.) degree in electrical and communication engineering from the Faculty of Engineering, International Islamic University, Malaysia, in 2016. He is currently an Associate Professor and the Head of the Network and Communication Technology Research Laboratory, Center for Cyber Security, Faculty of Information Science and Technology, Universiti Kebangsaan Malaysia (UKM). He specializes in cutting-edge information-centric networks: computer networks, data communication and security, mobile networks and privacy protection, cyber-physical systems, the Industrial IoT, transparent AI, and electric vehicle networks. He has published over 150 indexed papers in ranked journals and conference proceedings. He is a member of the Institution of Engineering and Technology and the Internet Society. He is a Certified Professional Technologist (P.Tech./Ts.) in Malaysia. He also served the IEEE Student Branch as the Chair, from 2014 to 2016. He has actively participated in many events, workshops, and trainings for Malaysia's IEEE and IEEE Humanity Programs. He works as an editorial member in many prestigious high-impact journals, such as IEEE, IET, Elsevier, Frontier, and MDPI; and as the general chair, the co-chair, and a speaker for conferences and workshops for the shake of society and academy knowledge building and sharing and learning. He has been contributing and working as a volunteer for underprivileged people for the welfare of society.

MOHAMMED AHSAN RAZA NOORI received the Graduate degree from Aligarh Muslim University, Aligarh, Uttar Pradesh, India, in 2005, and the M.C.A. degree from Uttar Pradesh Technical University, Lucknow, Uttar Pradesh, India, in 2009. He has more than nine years of teaching experience in computer science. His research interests include natural language processing, big data analytics, machine learning, deep learning, and optimization techniques.

ABDUL HADI ABD RAHMAN received the Ph.D. degree from UTM, in 2016, for his work on the perception of autonomous vehicle projects. He is currently an Associate Professor with the Machine Learning and Vision (MLV) Laboratory, Center for Artificial Intelligence Technology (CAIT), UKM. His research interests include artificial intelligence, autonomous robotics, and computer vision.

SHAYLA ISLAM (Senior Member, IEEE) received the B.Sc. degree in computer science and engineering from International Islamic University Chittagong, Bangladesh, the M.Sc. degree from the Department of Electrical and Computer Engineering, International Islamic University Malaysia (IIUM), in 2012, and the Ph.D. degree in engineering from the Electrical and Computer Engineering (ECE) Department, IIUM, in 2016, under Malaysian International Scholarship. She is currently an Associate Professor with UCSI University, Malaysia. She was awarded the Silver Medal for her research work with International Islamic University Malaysia. Consequently, she was awarded the Young Scientist Award for contributing a research paper at the second International Conference on Green Computing and Engineering Technologies 2016 (ICGCET'16), organized by the Department of Energy Technology, Aalborg University, Esbjerg, Denmark.



MOHAMMAD HALEEM received the B.Sc. degree (Hons.) in statistics from AMU, Aligarh, the M.C.A. degree from HBTI, Kanpur, and the M.Tech. degree in CSE and the Ph.D. degree in computer science from Integral University, Lucknow, Uttar Pradesh, India. He has 15 years of teaching experience. He is currently an Associate Professor with the Department of Computer Science, Era University, Lucknow, India. He has published more than ten research papers in reputed international journals and conferences. He is also a member of various professional societies. The research area is requirement engineering and requirement uncertainty data science.

SHEIKH FAHAD AHMAD received the B.Tech. and M.Tech. degrees in computer science and engineering from Integral University, Lucknow, Uttar Pradesh, India, in 2008 and 2013, respectively, and the Ph.D. degree in computer science and engineering from Babu Banarasi Das University Lucknow, Uttar Pradesh, in 2020. He is currently an Assistant Professor with the Department of Computer Science and Engineering, GITAM, and is deemed a university student in Hyderabad. He has more than 13 years of teaching experience in reputed universities. He is also a member of the editorial boards, program, and technical committees of many international journals and on the advisory committees of many international journals. He has around 30 international publications and three patents. His research interests include sound analysis, pattern recognition, machine learning, and data science.

ASIF KHAN (Member, IEEE) received the B.Sc. (Hons.) and Master of Computer Science and Application (M.C.A.) degrees from Aligarh Muslim University, India, and the Ph.D. degree (Hons.) in computer science and technology from the University of Electronic Science and Technology of China (UESTC), China, in 2016. He was an Adjunct Faculty Member at the University of Bridgeport, USA, for the China Program in summer 2016. Previously, he was a Visiting Scholar of big data mining and its application at the Chongqing Institute of Green and Intelligent Technology (CIGIT), Chinese Academy of Sciences, Chongqing, China. He was a Postdoctoral Scientific Research Fellow with UESTC. He is currently an Assistant Professor at Integral University, India. He contributes to many international journals with robotics and vision analyses about the contemporary world in his articles. His research interests include machine learning, robotics vision, and new ideas regarding vision-based information critical theoretical research. He received the UESTC Academic Achievement Award and the Excellent Performance Award in 2015 and 2016, respectively.

FATIMA RAYAN AWAD AHMED received the B.S. degree in computer science from Sudan University of Science and Technology, in 2004, and the M.S. and Ph.D. degrees in computer science from El Neelain University, Sudan, in 2007 and 2021, respectively. She joined to the Information Systems Department, Sattam University, Saudi Arabia, in 2013, as an Assistant Professor (2013–2016) and has been with the Computer Science Department, since 2017. In 2004, she joined the IT Department, Sudanese Company for Telecommunications, Sudan, as a Computer Programmer, where she analyzed, designed, and programmed a set of systems. Her research interests include artificial intelligence, systems and algorithms analysis and design, web applications, and E-learning.

NISSREIN BABIKER MOHAMMED BABIKER received the B.Sc. degree in computer science from Sudan University of Science and Technology, in 2004, the M.S. degree from Gazira University, in 2012, and the Ph.D. degree in information technologies from Sudan Academic University, Sudan, in 2018. She joined the Information Systems Department, Khalid University, Saudi Arabia, as a Lecturer (2013–2016) and has been the Information System Department, University of Bisha, since 2017. She worked in the IT Department, Ministry of Communications and Technologies (2004–2012), Sudan, where she analyzed, designed, programmed a set of systems. Her research interests include artificial intelligence, web applications, E-learning, the IoT, networks, cyber security, and blockchain.

THOWIBA E. AHMED received the B.S. degree in computer science and statistics from Al Neelain University, Sudan, in 2004, and the M.S. and Ph.D. degrees in information technology from Al Neelain University, Sudan, in 2007 and 2012, respectively. She has been with the Computer Science Department, College of Science and Humanities, Imam Abdulrahman Bin Faisal University, Saudi Arabia, since 2013, as an Assistant Professor, and was the Head of the Computer Science Department (2016–2019). In 2004, she joined the National Highway Authority, Sudan, as a Programmer and after that as an Information Network Supervisor. In 2004, she joined Al Neelain University as a Collaborative Teaching Assistant and a Collaborative Assistant Professor, in 2012, and in 2012, she joined Emirates College of Science and Technology, Sudan, as an Assistant Professor and the Head of the Information Technology Department. Her research interests include data science, expert systems, HCI, systems and algorithms analysis and design, web applications, and E-learning.

ATTA UR REHMAN KHAN (Senior Member, IEEE) is currently an Associate Professor with the College of Engineering and Information Technology, Ajman University, United Arab Emirates. Before joining AU, he was a Postgraduate Program Coordinator with Sohar University Oman, the Director of the National Cybercrime Forensics Laboratory, Pakistan, the Head of the Air University Cybersecurity Center, Pakistan, the Conferences Chair of IEEE Islamabad Section Pakistan, and the Research Committee Chair of King Saud University, Saudi Arabia. He has extensive experience in teaching, research, and industry at key positions. He has developed multiple degree programs, published research articles in reputed journals/conferences, and edited/co-authored multiple books. Some of his publications have appeared among the most popular articles of the IEEE ComSoc Society. They are among the top 1% highly cited articles of the academic field of computer science (citations received concerning publication year) as per Web of Science. A Stanford University report states, he is among the World's Top 2% Scientists. Currently, he is serving on the editorial boards of *Journal of Cluster Computing* (Springer), *IEEE Communications Magazine*, *Journal of Network and Computer Applications* (Springer), *The Computer Journal* (Oxford), IEEE SDN Newsletter, IEEE Access, *KSII Transactions on Internet and Information Systems*, and *Ad Hoc & Sensor Wireless Networks*. Moreover, he is a Program Evaluator of Pakistan's National Computing Education Accreditation Council (NCEAC); a member of Amazon Web Services (AWS) Educate Program and NVIDIA GPU Educators Program; steering committee member/PC chair/track chair/technical program committee (TPC) member of over 80 international conferences.